

---

# Fondamenti di Intelligenza Artificiale M

Altri possibili esercizi

## Esempi di possibili esercizi – Description Logic e SW e domande sui seminari in generale

---

- Introdurre brevemente la differenza tra Open World Assumption e Close World Assumption, illustrando tale differenza con una piccola knowledge base di esempio.
- Illustrare brevemente i possibili modelli semantici visti nel corso per rappresentare la conoscenza.
- Spiegare brevemente, nell'ambito dell'iniziativa Semantic Web, il ruolo degli standard RDF e OWL.
- Nell'ambito di Semantic Web, introdurre brevissimamente la funzione dello standard SPARQL, eventualmente con qualche esempio di query.
- Considerando gli operatori (EXISTS, FORALL, AND, FILLS) e i simboli logici di Description Logic visti a lezione, rappresentare la seguente conoscenza: “Le conferenze importanti sono tutte quelle che hanno almeno 3 iscritti. Le conferenze italiane sono quelle dove tutti gli iscritti sono italiani. Alla conferenza “xyz” si sono iscritti Federico, Marco e Paolo. Tutti e tre sono italiani.” Cosa è possibile inferire in Description Logic da questa base di conoscenza?

## Esempi di possibili esercizi – Meta-interpreti in Prolog

---

Semplicissime varianti del vanilla meta-interprete

## Esempi di possibili esercizi – Planning

---

È dato lo stato iniziale descritto dalle seguenti formule atomiche:

**[ontable(a,p1), ontable(d,p3), on(c,d), clear(a), clear(c), empty(p2), handempty]**

(a,c,d rappresentano dei blocchi e p1,p2,p3 sono le uniche 3 posizioni occupabili del tavolo)

Le azioni sono modellate opportunamente come segue:

- **pickup(X,Pos)**  
PRECOND: ontable(X,Pos), clear(X), handempty  
DELETE: ontable(X,Pos), clear(X), handempty  
ADD: holding(X), empty(Pos)
- **putdown(X,Pos)**  
PRECOND: holding(X), empty(Pos)  
DELETE: holding(X), empty(Pos)  
ADD: ontable(X,Pos), clear(X), handempty

(Pos è una variabile che rappresenta la posizione del tavolo, X,Y rappresentano blocchi)

## Esempi di possibili esercizi – Planning

---

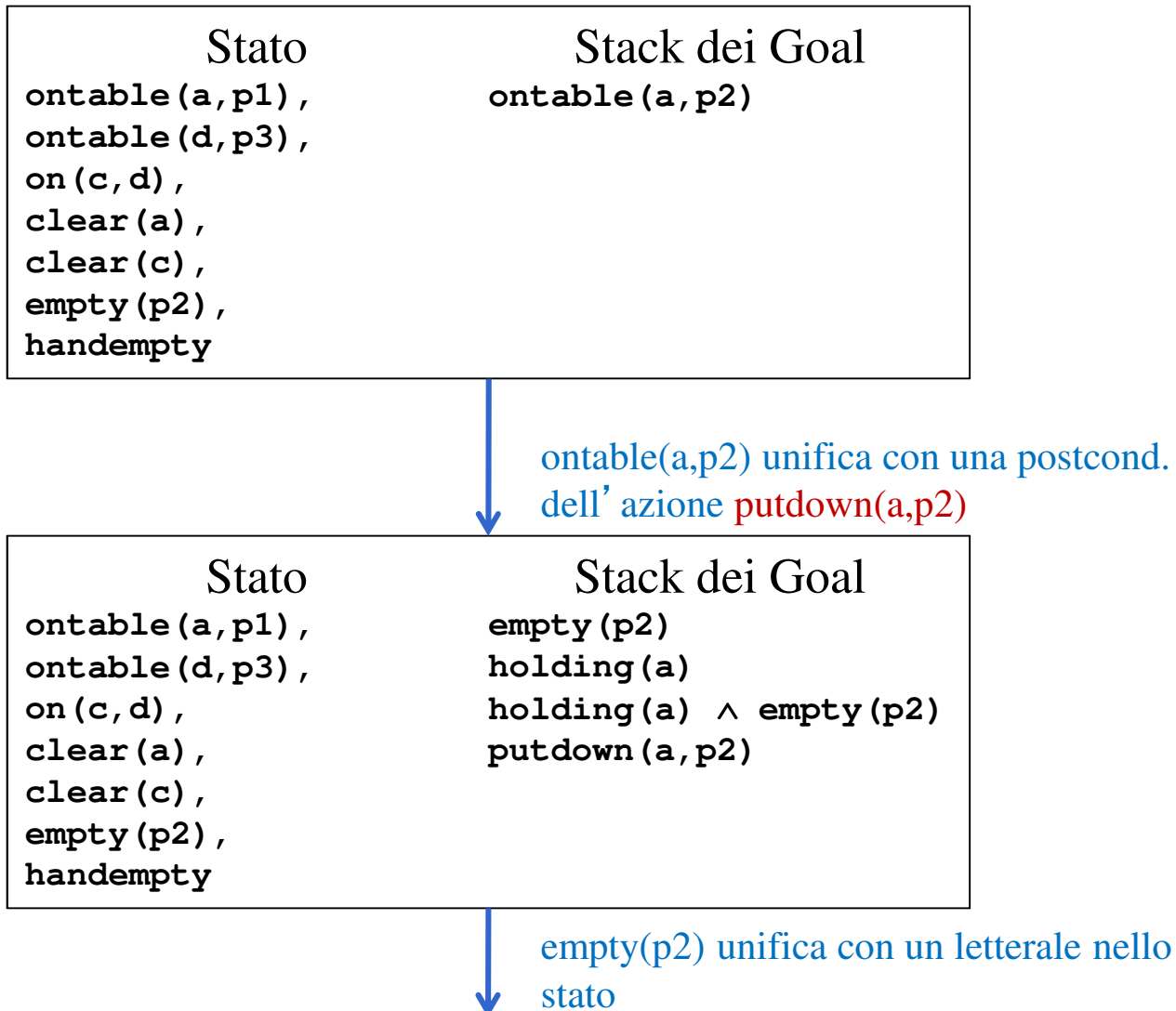
- **stack(X,Y)**  
PRECOND: holding(X), clear(Y)  
DELETE: holding(X), clear(Y)  
ADD: handempty, on(X,Y), clear(X)
- **unstack(X,Y)**  
PRECOND: handempty, on(X,Y), clear(X)  
DELETE: handempty, on(X,Y), clear(X)  
ADD: holding(X), clear(Y)

e il goal **ontable(a,p2)**

Si descriva come l' algoritmo lineare backward STRIPS trova un piano per questo goal (quindi si mostri SOLO una strada nell' albero di ricerca considerando come ordinamento tra precondizioni in and nello stack quello che considera prima le precondizioni che unificano direttamente con un letterale dello stato e poi quelle che per essere soddisfatte hanno bisogno di una azione). Si descriva lo stato e lo stack dei goal passo passo.

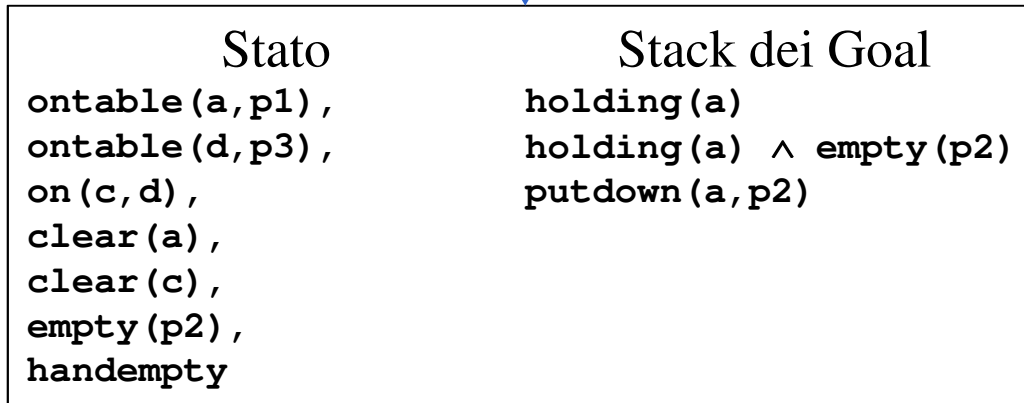
## Esempi di possibili esercizi – Planning

---

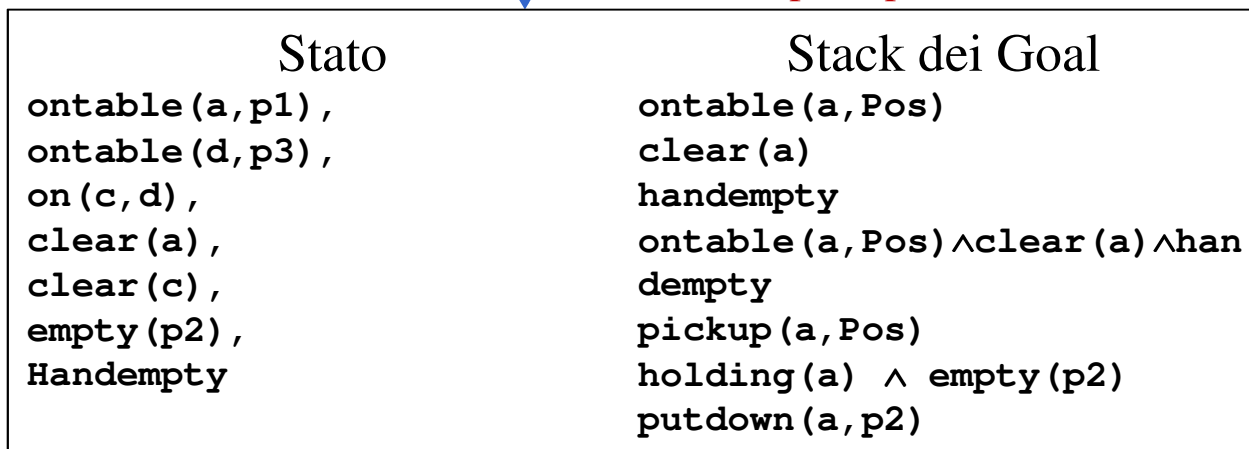


# Esempi di possibili esercizi – Planning

empty(p2) unifica con un letterale nello stato



holding(a) unifica con una postcond. dell'azione pickup(a,Pos)



ontable(a,Pos) unifica con un letterale dello stato Pos/p1

## Esempi di possibili esercizi – Planning

ontable(a,Pos) unifica con un letterale dello stato Pos/p1

Stato	Stack dei Goal
<code>ontable(a,p1), ontable(d,p3), on(c,d), clear(a), clear(c), empty(p2), Handempty</code>	<code>clear(a) handempty ontable(a,p1) ^ clear(a) ^ handempty pickup(a,p1) holding(a) ^ empty(p2) putdown(a,p2)</code>

clear(a) ed handempty unificano con un letterale nello stato

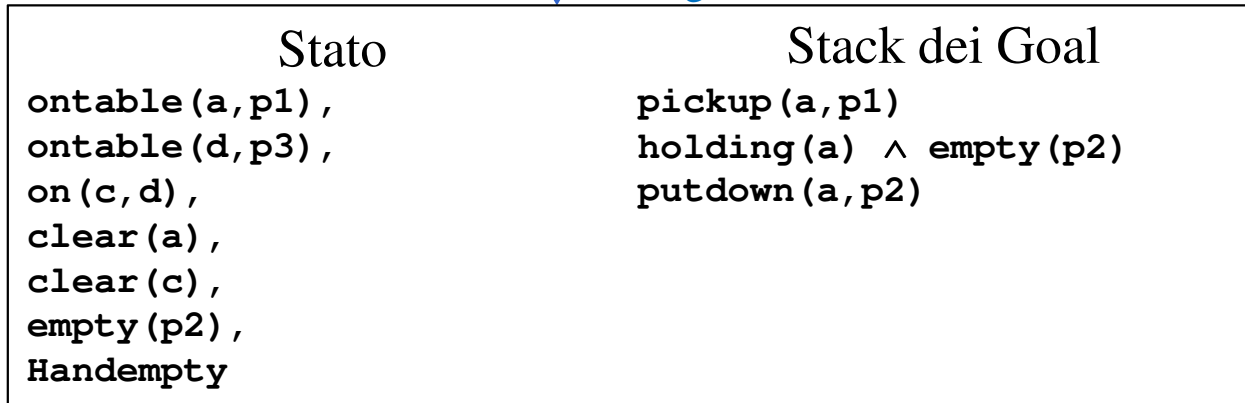
Stato	Stack dei Goal
<code>ontable(a,p1), ontable(d,p3), on(c,d), clear(a), clear(c), empty(p2), Handempty</code>	<code>ontable(a,p1) ^ clear(a) ^ handempty pickup(a,p1) holding(a) ^ empty(p2) putdown(a,p2)</code>

la congiunzione è soddisfatta

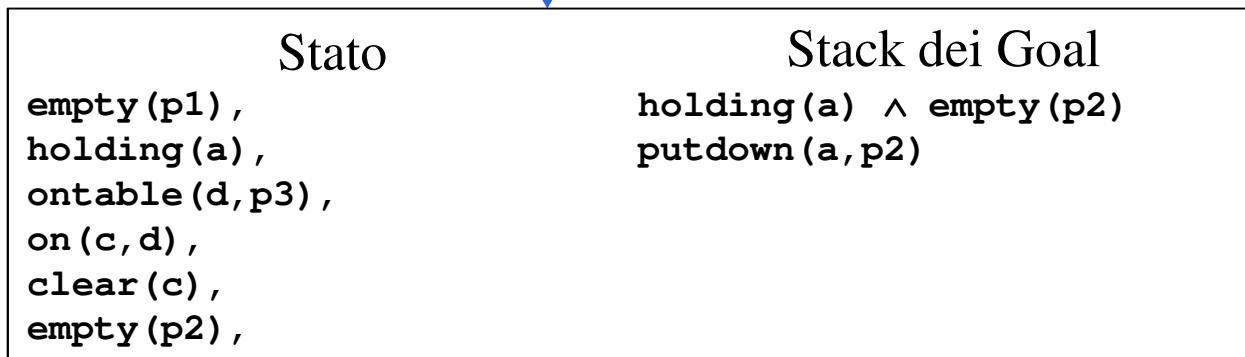


## Esempi di possibili esercizi – Planning

↓ la congiunzione è soddisfatta



↓ eseguo `pickup(a, p1)`



↓ la congiunzione è soddisfatta

## Esempi di possibili esercizi – Planning

↓ la congiunzione è soddisfatta

Stato	Stack dei Goal
<code>empty (p1) ,</code> <code>holding (a) ,</code> <code>ontable (d, p3) ,</code> <code>on (c, d) ,</code> <code>clear (c) ,</code> <code>empty (p2) ,</code>	<code>putdown (a, p2)</code>

↓ Eseguo `putdown(a,p2)`

Stato	Stack dei Goal
<code>empty (p1) ,</code> <code>handempty (a) ,</code> <code>ontable (d, p3) ,</code> <code>on (c, d) ,</code> <code>clear (c) ,</code> <code>ontable (a, p2)</code>	

# Planning

---

- Oppure domande di teoria o modellazione di azioni in STRIPS